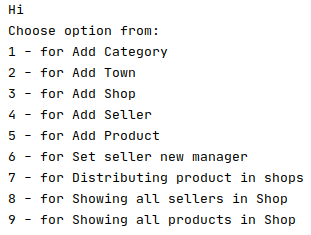
# Spring Boot Introduction Lab

## Functionality Overview

The application should be able to easily **accept** hard-formatted. The application is called – **Super Market**.

Look at the pictures below to see what must happen:

* When you've ran the project:



* Successfully adding a new category:

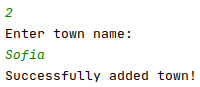
Category1

C:\Users\Primo\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Category2.png

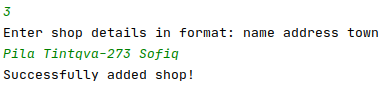
* Invalid category name message:

Cat3

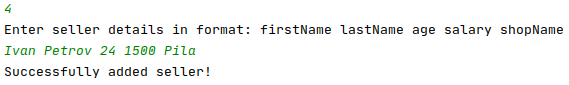
* Successfully adding a new town:

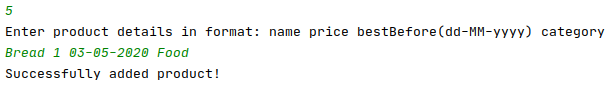


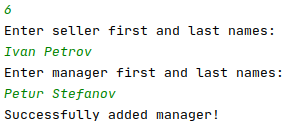
* Successfully adding a new shop:

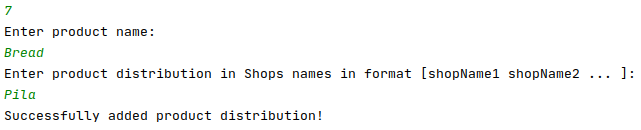


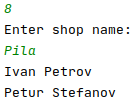
* Successfully adding a new seller:



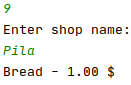
* Successfully adding a new product:  
  
* Successfully setting a manager:



* Successfully distributing products to shops:
* Showing all sellers in a specific shop:



* Showing all products in a specific shop:



## Model Definition

There are 5 main models that the **Super Market database** application should contain in its functionality.

Design them in the **most appropriate** way, considering the following **data constraints**:

### Town

* id – a **char sequence**
* name – a **char sequence**

### Shop

* id – a **char sequence**
* **address –** a **char sequence.** It's **unique** andcannot be **null.** Must be at least **2 characters.**
* **name –** a **char sequence.** Must be at least **2 characters.**

### Seller

* id – a **char sequence**
* firstName – a **char sequence.** Cannot be **null.** Must be at least **2 characters.**
* lastName – a **char sequence.** Cannot be **null.** Must be at least **2 characters.**
* **age –** an **integer.** Cannot be **null**. The person must be at least 18 years old.
* **salary –** a **number** (must be a positive number). Cannot be **null.**

### Category

* id – a **char sequence**
* name – a **char sequence.** It's **unique** andcannot be **null.** Must be at least **2 characters.**

### Product

* id – a **char sequence**
* bestBefore – a **date**
* **description – a very long char sequence**
* **name –** a **char sequence.** Cannot be **null.** Must be at least **2 characters.**
* **price –** a **number** (must be a positive number). Cannot be **null.**

**NOTE**: Name the entities and their class members, **exactly** in the **format stated** above. Do not name them in snake case with the dashes, of course.

#### Relationships

Your partners gave you a little hint about the more complex relationships in the database, so that you can implement it correctly.

One Shop may be in only one Town, and one Town may have many Shops.

One **Seller** may be in only one **Shop**, and one **Shop** may have many **Sellers**.

One **Product** may have only one **Category**, and one **Category** may have many **Products**.

One **Product** may be in many **Shops**, and one **Shop** may have many **Products.**

